CHARACTERIZING HOW JEPA CONSTRUCTS REPRESENTATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

In joint-embedding predictive architectures (JEPA), an encoder and a predictor are trained together to produce a representation that minimizes the error in predicting the representation of the next observation. Because the representations of both the current and the next observation are learned, it is unclear which features they will include. Earlier works portray JEPA as selecting features based on their individual predictability, raising the concern that control-relevant features may be omitted if they are hard to predict. To address this concern and better understand how JEPA selects which features to encode, we propose the **predictive closure** characterization. Based on a reformulation of the JEPA training objective, we show that selection depends on *joint* properties of features. We support this characterization with experiments showing that unpredictable features can become better represented when they help predict other features, and that changing a feature's predictive relationships can affect how well others are represented. We also demonstrate implications for steering feature selection: adding an auxiliary objective targeting one feature can strengthen the representation of others, and supplying a feature as side information to the predictor can change how well other features are represented. We hope this work helps practitioners understand and steer JEPA representations and guides researchers in improving how these systems learn.

1 INTRODUCTION

We study how JEPA selects which features to encode. A joint-embedding predictive architecture (JEPA) aims to learn representations useful for planning and comparing observations, without explicitly specifying which features those representations should include (LeCun, 2022). JEPA consists of an encoder that maps each observation (for example, three consecutive frames from a video) to a latent representation and a predictor that uses the current observation's representation to forecast the representation of the next observation. The training objective combines a prediction loss with regularization designed to prevent the latent representation from collapsing to the same point for every input. Although JEPA-based systems have shown strong performance across a range of applications (Assran et al., 2025; Maes et al., 2026; Rao et al., 2026; Larey et al., 2026; Zhu & Choromanska, 2026; Kuhn et al., 2026), it remains unclear how JEPA selects the features encoded in its latent representation. This is important because this representation acts as the world model JEPA builds for each observation (García et al., 1989; Schmidhuber, 1990; Sutton, 1991; Ha & Schmidhuber, 2018), so features absent from it cannot inform comparisons between observations or planning, even if predictions in that latent space are perfectly accurate.

JEPA learns what to predict together with how to predict it. The encoder determines both sides of the prediction problem: the present embedding is the predictor's input, and the future embedding is its prediction target. This makes it difficult to reason about which features JEPA will encode. Consider our running example of a traffic light (Figure 1). The light's next state is not predictable from external signals, so leaving it out could make the future representation easier to predict. Yet its current state helps predict whether the cyclist will be riding or stopping next, so leaving it out could also make that same representation harder to predict. Should JEPA retain the light despite its own unpredictability?

In recent work, Pendharkar (2026) argues that features such as traffic-light that are unpredictable will not be represented. This is consistent with other works that explain feature retention in terms

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

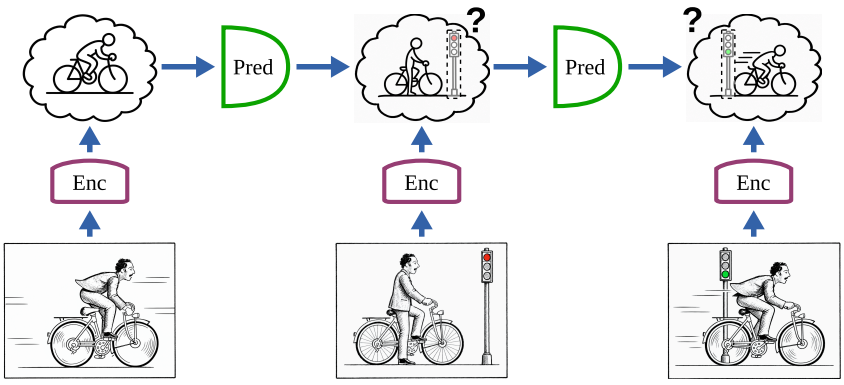


Figure 1: **Will the traffic light be included in the latent representation?** The bottom row shows successive observations; Enc and Pred denote the encoder and predictor. The light's color is chosen at random in each observation and cannot be predicted from earlier observations, yet its current value helps predict whether the cyclist will be riding or stopping next. Reasoning about this choice can feel like chasing one's own tail: what is useful to represent depends on what must be predicted, while what must be predicted depends on what is represented. Cyclist artwork adapted from Scott McCloud's *Understanding Comics* (1993).

of a feature's own temporal behavior, such as its rate of change (Sobal et al., 2022) or predictability (Littwin et al., 2024; Team, 2026; Karmann et al., 2026). However, determining whether to include a feature individually does not capture how that choice depends on which other features are represented.

In this paper, we argue for a characterization that takes into account the dependencies between features. We reformulate JEPA's training objective in terms of the *joint* predictive value of feature sets (Section 3). When the latent cannot retain all observation features, the objective favors, among the sets the encoder can represent, the set with the greatest mutual information between the joint values of its features at time t and those of the same features at time $t + 1$. This reveals that the traffic light would be included if it belongs to such a maximizing set, even if the traffic light itself is impossible to predict. We call the selected set of features a **predictive closure**, and use the same term for our characterization of feature selection in JEPA.

We use this characterization to reason about feature selection by comparing sets (Section 3.2), then validate its predictions experimentally (Section 4). We construct image sequences with controlled predictive relations and use held-out probes to measure feature decodability from representations of JEPA systems trained on those sequences. First, we find that an unpredictable traffic-light-like feature becomes more decodable as its usefulness for predicting other features increases. Second, strengthening one predictive connection within a set of features that help predict one another can increase its members' representation while reducing that of a competing set whose predictive relations are unchanged. Third, even a highly predictive set of features can remain poorly represented if it requires more capacity than the latent provides. Together, these findings support a characterization of feature selection in terms of sets, with selection depending on both the predictive value of a set and the amount of information it contains.

Joint selection further suggests that targeting one feature during training may change the representation of its predictive partners and competitors. We investigate these effects through an auxiliary objective (LeCun, 2022; Destrade et al., 2025; Zhu et al., 2026; Vujinovic & Kovacevic, 2026) and by supplying a feature directly to the predictor (Section 5). We find that adding an auxiliary loss to reconstruct a single image can strengthen the representation of its untargeted predictive partner while reducing that of competing feature sets. We also find that supplying a feature directly to the predictor strengthens the representation of some features while weakening others. This suggests using side information extracted from observations to steer feature selection, even when that information is not an action (Section 5.2.1).

Predictive closure offers a way to reason about which features JEPA represents and how changes to training can influence this selection. We hope this view helps practitioners and researchers understand these systems and steer them toward features relevant to downstream tasks.

1.1 Background: JEPA's Training Objective

A joint-embedding predictive architecture (JEPA) learns a representation by predicting the latent representation of the next observation. As illustrated in Figure 1, JEPA includes an encoder f that maps an observation x_t at time t to a latent representation $f(x_t)$, and a predictor g that takes $f(x_t)$ as input to predict $f(x_{t+1})$. Training combines a prediction loss on observation pairs with a regularization loss on the latent distribution (LeCun, 2022):

$$\mathcal{L}_{\text{JEPA}} \triangleq \|g(f(x_t)) - f(x_{t+1})\|_2^2 + \lambda \mathcal{R}(f(x_t)). \tag{1}$$

The parameter $\lambda > 0$ controls the weight of the regularizer $\mathcal{R}$, which acts on the distribution of $f(x_t)$ and encourages it to match a chosen distribution to prevent collapse (such as the distributions proposed in Balestriero & LeCun (2025); Wu et al. (2026); Kuang et al. (2026); Nicollier et al. (2026)).

These latent predictions can also support planning: a planner compares predicted future representations under candidate action sequences to choose how to act. In this setting, the predictor receives an action a_t alongside the current latent, giving $g(f(x_t), a_t)$ (Maes et al., 2026; Caselli et al., 2026). We discuss how this additional input affects feature selection in Section 5.2.

2 Preliminaries

We use features to describe observations through identifiable properties, such as the traffic light's color or whether the cyclist is riding or stopping. This lets us characterize what JEPA's encoding retains in terms of those properties and their predictive relationships. We measure this retention through recoverability: a feature is represented when a reader with bounded sensitivity can recover it from the latent for use by the predictor or a downstream model. We assume that the latent lacks the capacity to represent all observation features simultaneously, so our question is which features it retains.

Features. A feature ϕ_i describes a property of an observation, such as the traffic light's color or the cyclist's speed. Formally, ϕ_i is a mapping from observations to scalar values, and we write $y_t^i = \phi_i(x_t)$ for its value at time t. This excludes properties that cannot be inferred from the observation, such as which watch the cyclist wears when it is completely hidden and no other clues reveal it. Different mappings define different features. For example, a mapping that matches whether the cyclist is riding or stopping on some observations but not others defines a different feature. A **feature set** $S = \{\phi_1, \ldots, \phi_m\}$ is a collection of these mappings. We write $Y_t(S) = (\phi_i(X_t))_{\phi_i \in S}$ for their values on an observation X_t at time t.

Represented feature sets. We say that a feature ϕ_i is **represented** in the latent $f(x_t)$ when a reader d_i with bounded sensitivity recovers its value across observations: $d_i(f(x_t)) = y_t^i$ for all t (formalized in Appendix A.2). For example, the cyclist's riding-or-stopping state is represented if a reader can recover it from the latent. We write $S(f)$ for the complete set of features recoverable in this way from the latent produced by encoder f.

Fixed latent entropy. Representing a feature requires capacity and we assume that all encoders match the same Gaussian target prescribed by regularization at each time, with P denoting a finite-precision latent distribution. [1] The shared distribution fixes the Shannon entropy of the latent distribution $H(P)$, which we refer to as the latent's **entropy constraint**. We assume that the entropy constraint $H(P)$ is less than the observations' entropy so not all features can be represented. We call a feature set S **feasible** when it is the complete set represented by some encoder f, so that

[1] All encoders use the same numerical precision, and readers access the stored values. This idealizes the distribution matching encouraged by regularizers such as SIGReg, for which Balestriero & LeCun (2025, Section 4.3) provide empirical support.

$S = S(f)$. The values of the complete represented feature set and the latent determine one another, consequently, every feasible set satisfies $H(Y_t(S)) = H(f(X_t)) = H(P)$: its feature values have the same joint entropy (see more details in Appendix A.2). If we could directly choose which features are represented, adding a feature that provides new information would require removing others to keep the resulting set feasible.

3 PREDICTIVE CLOSURE

With these definitions, our question about what JEPA represents becomes a question about which features belong to $S(f)$ for the learned encoder. We reformulate the training objective to characterize this set.

Prediction loss and information. Under the fixed latent distribution and a Gaussian prediction-error assumption, we show that minimizing expected squared prediction error is equivalent to maximizing the mutual information between consecutive representations. We denote any such optimal encoder by $f^\star$:

$$f^\star \in \arg\min_f \min_g \|g(f(X_t)) - f(X_{t+1})\|_2^2 = \arg\max_f I\big(f(X_t); f(X_{t+1})\big). \tag{2}$$

The squared error in Equation (2) denotes its expectation over consecutive observation pairs. The assumption and proof are detailed in Appendix A.1. The constrained objective therefore favors latent representations that share the most information across successive observations, subject to the fixed distribution imposed by regularization.

From representations to feature sets. Because the complete represented feature set preserves the information in the latent, we can express this objective directly in terms of the set whose membership we seek:

$$\boxed{S^\star := S(f^\star) \in \arg\max_{S \text{ feasible}} I\big(Y_t(S); Y_{t+1}(S)\big).} \tag{3}$$

The simple derivation appears in Appendix A.2. We call such a maximizing set a **predictive closure**: among feasible feature sets, it has the greatest mutual information between the joint values of its features at time t and the joint values of those same features at time $t+1$. We call this mutual information $I_{\text{pred}}(S) = I(Y_t(S); Y_{t+1}(S))$ the set's **predictive information**.

Intuitively, given a set of features in JEPA's latent, an additional feature is a candidate for inclusion if its value at time t helps predict the values of features in the set at time $t+1$, or if its value at time $t+1$ can be predicted from the values of features in the set at time t.

More generally, features are assessed jointly through the sets in which they can be included.[2] This means that neither how easily a feature can be predicted nor how much information it provides determine by themselves whether it belongs to the predictive closure. **A feature ϕ will be represented if it is a member of some feasible feature set S_ϕ with greater predictive information than any other feasible set.** These competing sets can differ in many features besides ϕ, and some may share only constant features. Whether one part of an observation is represented can therefore depend on whether another part is represented, even when neither helps predict the other.

3.1 EXPLAINING EARLIER FINDINGS

Predictive closure helps explain earlier empirical findings on which features JEPA-based systems encode. Sobal et al. (2022) found that a moving object was better represented when its background was independently resampled between observations than when it remained fixed within each sequence. We explain this by the high predictive information of sets containing those fixed-background features, which resampling the background removes: the background's features then contribute no predictive information, so the encoder will favor sets containing the moving object's features. Similarly, Pendharkar (2026) showed that an independently resampled visual cue, appearing over a predictably changing background, was barely decodable from JEPA's latent despite determining which action earned a reward (reward was not included in the training objective). We explain this by the cue's features adding no predictive information to any set, despite their relevance to reward.

[2] In optimization terminology, this is generally a non-additive, or non-modular, set objective: its value depends on the combination of selected features rather than a sum of fixed individual contributions.

3.2 PREDICTING FEATURE SELECTION

Predictive closure also allows us to make predictions about the feature-selection behavior of JEPA encoders, including in the traffic-light case. We now present five such predictions.

Cross-subsidization. An unpredictable feature ϕ will be represented if it is a member of the feasible set with the greatest predictive information. This can be achieved if ϕ is highly predictive of other represented features, which subsidizes the predictive information "sacrificed" by its own unpredictability. Hence, we predict that **an unpredictable feature will be represented if it is sufficiently predictive of other represented features**. This means that for scenarios like the one depicted in Figure 1, where the cyclist's features are determined by the traffic light and the light itself is unpredictable, the traffic light will be represented.

Set switching. Characterizing feature selection as set selection also implies that the **predictive information of one set of features can impact the representation of a disjoint set**. Consider a scenario where the observations contain two feature sets with predictive relationships within each set, but not between them, and the latent has the capacity to fully encode either set but not both. Increasing[3] the strength of the relationships within one set sufficiently will lead to it being represented over the second set, even if the second set remains unchanged. For example, adding sensors for an independent, highly predictable process can displace features of the original observations from the latent.

Feasibility dependence. A feature set can have greater predictive information than every feasible alternative and still not be represented, since a set whose entropy exceeds the latent capacity is not feasible. Thus, we predict that **a feature will not be represented if it only belongs to high-predictive-information sets whose entropy exceeds the latent capacity**. For example, enlarging a set of mutually predictive features can make it infeasible under a fixed latent capacity, leading to an entirely different set being represented.

Implications for steering with an auxiliary loss. When adding an auxiliary loss that requires some features, we force those features into the predictive closure. We predict that **fixing any features in the representation will impact the selection of the remaining features**, such that they maximize predictive information of the set including the fixed features. The included features will be not only those that the auxiliary loss requires, but also those that have high predictive information with them. For example, an auxiliary loss for reconstructing the wind measurements can favor representation of a flag's state.

Implications for steering with side information. When supplying the predictor with side information A_t that it is not trained to predict (as is usually done for actions), the corresponding feature selection objective becomes maximizing $I((Y_t(S), A_t); Y_{t+1}(S))$. This extends the prediction-error assumption to condition on both predictor inputs. Adding A_t can therefore favor feature sets S whose future values A_t helps predict together with S. We predict that adding **side information can impact feature selection, without requiring the side information itself to be encoded in the latent**. For example, supplying wind measurements to the predictor can favor representation of a flag's state.

We validate the first three predictions in Section 4 and the two steering predictions in Section 5.

4 VALIDATING THE PREDICTIONS OF PREDICTIVE CLOSURE

We validate the first three predictions from Section 3.2 using image sequences with controlled predictive relationships. We approximate whether a target feature is represented by measuring how accurately a trained probe recovers its value from the latent on held-out observations.

Experimental setup. All experiments train JEPA on observation sequences of CIFAR-10 image collages (Krizhevsky, 2009) and evaluate specific *target features* using held-out probes of the frozen

[3]When we change properties of the observations we consider the counterfactual case where a JEPA system would be trained on the sequence after the change.

latent. These targets are the traffic-light state or binary indicators of target-image presence. We approximate how well features of the image at each position are represented by averaging accuracy across the binary target-image detection tasks associated with that position. We use representation of an image as shorthand for representation of its features. We use a three-dimensional latent representation and a training objective combining the prediction loss and SIGReg regularization with coefficients of 0.9 and 0.1, respectively. We choose this small latent to study competition between features under a restrictive capacity constraint. Unless otherwise specified, we train each configuration with three optimization seeds and report mean decoding accuracy, with error bars indicating standard errors across seeds. Unless stated otherwise, our findings are not specific to a single choice of λ (see sensitivity analysis in Appendix D) and additional training and probing details are given in Appendix C. We validate the predictive performance of the traffic-light and two-pair trained systems in Appendix E.

4.1 Cross-subsidization: representing unpredictable features

Scenario. We place a two-valued traffic light alongside two images in a collage. We write A and B for the images and L for the light's value, green or red. At every transition, exactly one image is replaced by a newly sampled image while the other persists. With probability q, the current light L_t determines which position is resampled; otherwise, the position is chosen uniformly (Figure 2, left). Green selects A for replacement and red selects B. The next light L_{t+1} is an independent uniform draw, so the future of the light is unpredictable at every value of q, while increasing q makes L_t more useful for predicting which images will persist. Because the shared encoder processes every frame, retaining the current light also places its unpredictable successor in the prediction target.

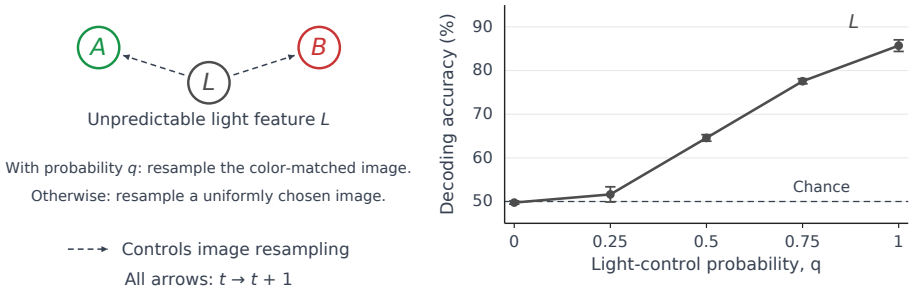


Figure 2: **Cross-subsidization: the unpredictable light becomes more decodable when it helps predict the images.** Increasing q makes the current light more useful for predicting which images persist, while its future remains unpredictable. Dashed line: 50% chance accuracy.

Results. As control probability q increases, the light feature L becomes better represented even though its future is impossible to predict (Figure 2). This shows that features are not selected solely according to how easy they are to predict. An unpredictable target feature can become more decodable when it helps predict other features, consistent with predictive closure. The trained systems also achieve lower latent prediction loss as q increases. Note that the transition in this scenario involves a nonlinear interaction between the light and the current images, which cannot be captured by frameworks that assume the predictor is a linear function such as Littwin et al. (2024).

4.2 Set Switching

Scenario. The two-pair setup places images A/B and C/D in a 2×2 collage. Within each pair, an image can be copied to the other position in the next frame; otherwise, a new image is sampled independently. We vary only the probability p_D that C_t is copied to D_{t+1} while keeping all other copying probabilities fixed (Figure 3, left; Appendix C.2).

Results. Increasing p_D shifts representation from features of A/B toward features of C/D, although the predictive relationships within A/B remain unchanged (Figure 3). Strengthening a predictive relationship within one pair can therefore affect representation in the other pair, even though

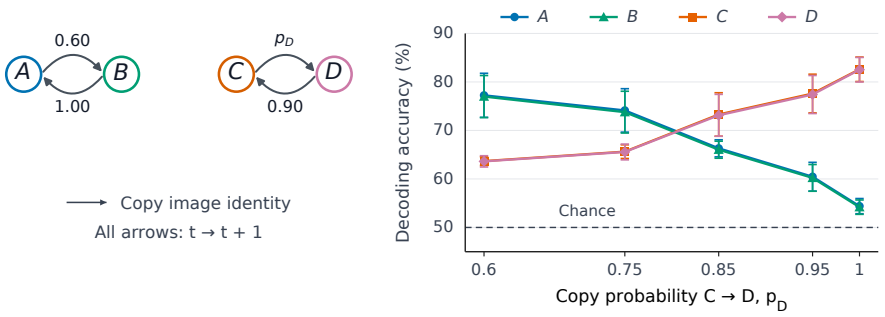


Figure 3: **Set switching: changing predictive relationships reverses which pair is more strongly represented.** Increasing p_D strengthens decoding of C/D and weakens A/B, whose predictive relationships remain fixed; neither pair predicts the other. Dashed line: 50% chance.

neither pair predicts the other. We also find similar decoding accuracy within each pair, consistent with the set-level characterization.

4.3 Feasibility Dependence

Scenario. To test feasibility dependence, we keep A/B and the latent size fixed and increase the competing cycle to $N \in \{2, 4, 8, 16\}$ images (Figure 4, left). Average within-cycle copying probabilities remain unchanged, so retaining the larger cycle requires representing features of more images (Appendix C.2).

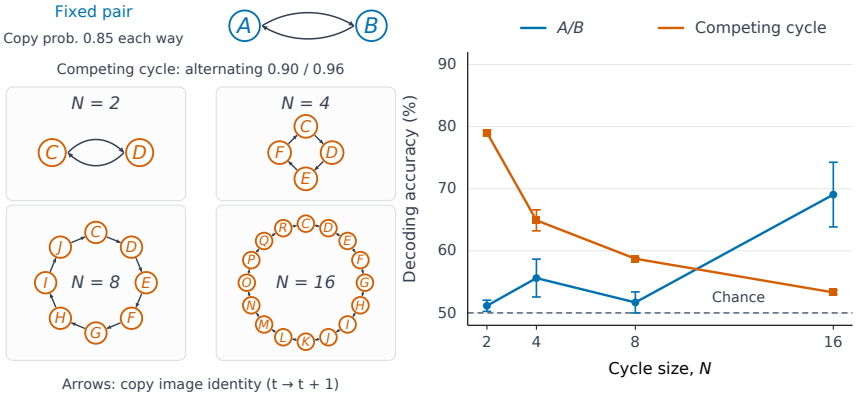


Figure 4: **Feasibility dependence: adding images to a cycle can reverse which images' features are more strongly represented.** Mean decoding accuracy across the competing cycle's positions decreases as N grows, despite unchanged average copying probabilities; at $N = 16$, the fixed A/B pair is better represented. Dashed line: 50% chance.

Results. Mean decoding accuracy for features of the competing images decreases at every increase in N in all three optimization seeds, despite unchanged average within-cycle copy probabilities (Figure 4). At $N = 16$, features of A/B are better represented in all three seeds; this endpoint reversal also holds for the worst and best positions in each cycle (Appendix B). We find that the increase in representation of A/B's features depends on λ (see Appendix D).

5 Validating Predictions for Steering

We now test the two steering predictions from Section 3.2. Since selection operates jointly, we examine how interventions targeting features of a single image affect both their predictive partners and

competing features. We consider two interventions: rewarding reconstruction of one image and supplying it as an additional predictor input. Both experiments use the two-pair setup from Section 4.2, keeping the sequence's predictive relationships, image pool, three-dimensional main latent representation, and training budget unchanged. We measure image-detection accuracy from the frozen main encoder alone, training probes on collages of images held out from encoder training and evaluating them on separate collages. Chance decoding accuracy is 50% for each feature individually.

5.1 An Auxiliary Reconstruction Objective

Auxiliary objectives have been proposed to steer JEPA representations toward task-relevant information. An auxiliary reconstruction objective encourages the encoder to retain features of the image being reconstructed. Section 3.2 predicts that this incentive will also favor their predictive partners, potentially at the expense of competing features.

Intervention. To test this prediction, we add an auxiliary objective that reconstructs a coarse version of the current C image from the main latent representation; only C is a reconstruction target, and the decoder is discarded before probing. We vary the reconstruction weight β across all five original values of p_D, using one optimization seed. The nonzero weights come from a preliminary search with both C and D as reconstruction targets (Appendix C.3).

Results. Rewarding reconstruction of C alone can make D's features more strongly represented and A/B's less so (Figure 5). This supports the prediction that steering can affect representation beyond the directly targeted features.

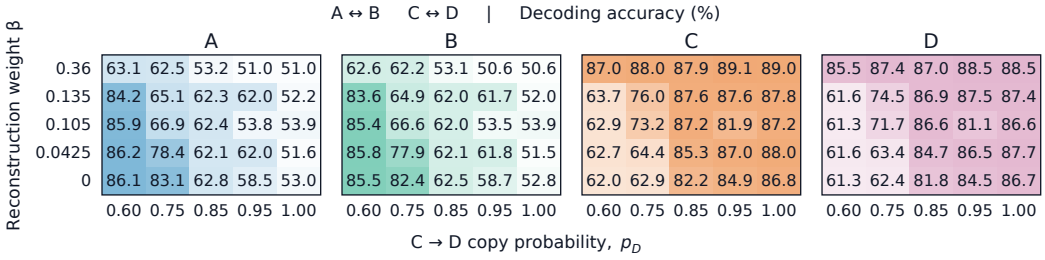


Figure 5: **Reconstructing one image can strengthen representation of its predictive partner.** The changes are not uniformly monotonic, and the baseline already favors C/D at high p_D. Each cell uses one optimization seed; all panels use the same 50–90% color scale.

5.2 Predictor Side Information

Supplying information directly to the predictor changes how well it can predict different features. Section 3.2 predicts that this can favor representation of features whose futures the additional information helps predict, with effects extending to competing features.

Intervention. To test this prediction, we supply the current C image through an additional predictor input, without an auxiliary loss or masking it from the main input. A separate encoder with the same architecture as the main encoder maps C to a three-dimensional vector, which we concatenate with the main latent representation before prediction; this adds encoder capacity while leaving the main representation's dimension unchanged. The side encoder is trained jointly with the main encoder and predictor through the prediction loss; SIGReg is applied only to the main latent representation. We evaluate all five values of p_D, paired with their no-side-input controls. We validate these systems' predictive performance in Appendix E.

Results. Supplying C changes how well features of the other images are represented, but produces a different redistribution from reconstruction (Figure 6). For $p_D = 0.75, 0.85, 0.95$, features of D become more strongly represented and those of B less represented in every seed, while representation of A's features increases throughout the sweep. These changes support the steering prediction,

but do not uniformly replace the A/B pair by C/D. Low decoding accuracy from the main latent alone does not establish that a feature is unavailable to the predictor: information about C or other features may be recoverable jointly from the main and side representations.

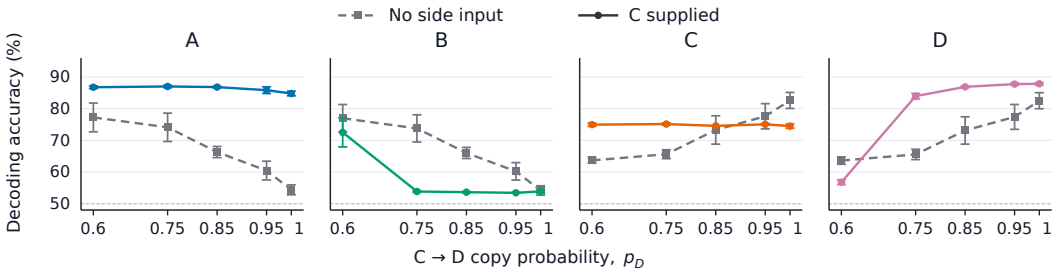


Figure 6: **Supplying C to the predictor changes how other features are represented.** Data and probe tasks are matched; probes use only the main latent. D becomes less represented at the lowest p_D and C at the highest. Dotted line: 50% chance.

5.2.1 Implications for representation learning

These results suggest that the choice of predictor inputs can influence feature selection, with implications for both routine action conditioning and deliberate steering of the learned representations.

Action conditioning. In action-conditioned JEPA, the predictor receives the agent's action alongside the current latent. Knowing the action may favor features whose futures it makes more predictable, together with their predictive partners, potentially strengthening control-relevant representation. Our observation-derived intervention illustrates the trade-off: at intermediate values of p_D, supplying C strengthens D but weakens B, potentially helping tasks that need D at the expense of those that need B. The consequences of genuine action conditioning remain to be tested.

Steering through extracted information. A practitioner can instead choose side information specifically to steer predictive closure. If a desired set of predictive features is poorly represented and one of its features can be extracted from the observation, supplying that feature to the predictor may strengthen representation of its predictive partners. Our intervention illustrates this possibility: supplying C can strengthen representation of D without directly supervising D. The mixed changes across the remaining features show why the intervention should be evaluated against all features needed downstream. Predictive closure thus suggests choosing a side input for its predictive relationships to a desired feature set, using access to one feature to influence representation of others.

6 Conclusion

We characterize JEPA representation learning through the joint predictive value of feature sets. We reformulate the training objective as selecting a **predictive closure**: a feasible feature set that maximizes mutual information between the joint values of its features at time t and those of the same features at time $t+1$. We provide empirical evidence to support this view: an unpredictable feature becomes more decodable when it helps predict others, and changing a competing set can weaken a feature's representation even when its own predictive relations are unchanged. Experiments with auxiliary objectives and predictor side information further show that targeting one feature can change how other features are represented and illustrate how this view can inform the design of JEPA systems to better represent task-relevant features.

This set-level view may also be relevant to models of biological representation learning that link representation to prediction (Palmer et al., 2015; Chalk et al., 2018; Singer et al., 2023).

We hope predictive closure provides a clearer way to reason about which features JEPA represents and informs the design and use of these systems in downstream tasks.

AI USE STATEMENT

We used generative AI to assist with coding and to polish parts of this paper. The authors take responsibility for the manuscript, code, and results.

REPRODUCIBILITY STATEMENT

We provide theoretical assumptions and derivations, experimental settings, and evaluation protocols in the paper and appendices. The supplementary materials include code and configurations for reproductions of all five main-text experiments.

REFERENCES

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025. URL https://arxiv.org/abs/2506.09985.

Randall Balestriero and Yann LeCun. LeJEPA: Provable and scalable self-supervised learning without the heuristics. *arXiv preprint arXiv:2511.08544*, 2025. URL https://arxiv.org/abs/2511.08544.

Niccolò Caselli, Francesco Massafra, Samuele Punzo, Salvatore Lo Sardo, Ippokratis Pantelidis, and Sathya Kamesh Bhethanabhotla. Mind the gap: Promises and pitfalls of hierarchical planning in LeWorldModel. *arXiv preprint arXiv:2607.12547*, 2026. URL https://arxiv.org/abs/2607.12547.

Matthew Chalk, Olivier Marre, and Gašper Tkačik. Toward a unified theory of efficient, predictive, and sparse coding. *Proceedings of the National Academy of Sciences*, 115(1):186–191, 2018. doi: 10.1073/pnas.1711114115.

Matthieu Destrade, Oumayma Bounou, Quentin Le Lidec, Jean Ponce, and Yann LeCun. Value-guided action planning with JEPA world models. *arXiv preprint arXiv:2601.00844*, 2025. URL https://arxiv.org/abs/2601.00844.

Carlos E. García, David M. Prett, and Manfred Morari. Model predictive control: Theory and practice – a survey. *Automatica*, 25(3):335–348, 1989. doi: 10.1016/0005-1098(89)90002-2. URL https://doi.org/10.1016/0005-1098(89)90002-2.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018. URL https://arxiv.org/abs/1803.10122.

Markus Karmann, Shile Li, Christian Internò, Bruno Andreis, David Klindt, Randall Balestriero, Jindong Gu, Philip Torr, Qi Zhang, Peng-Tao Jiang, Hao Zhang, Bo Li, and Onay Urfalioglu. Motionjepa: Preventing temporal feature collapse by capturing visual changes in latent space, 2026. URL https://arxiv.org/abs/2609.23881.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Yilun Kuang, Yash Dagade, Deep Chakraborty, Erik Learned-Miller, Randall Balestriero, Tim G. J. Rudner, and Yann LeCun. Radial-VCReg: More informative representation learning through radial gaussianization. *arXiv preprint arXiv:2602.14272*, 2026. URL https://arxiv.org/abs/2602.14272.

Lukas Kuhn, Giuseppe Serra, and Florian Buettner. Non-contrastive vision-language learning with predictive embedding alignment. *arXiv preprint arXiv:2602.00653*, 2026. URL https://arxiv.org/abs/2602.00653.

Ariel Larey, Elay Dahan, Amit Bleiweiss, Raizy Kellerman, Guy Leib, Omri Nayshool, Dan Ofer, Tal Zinger, Dan Dominissini, Gideon Rechavi, Nicole Bussola, Simon Lee, Shane O'Connell, Dung Hoang, Marissa Wirth, Alexander W. Charney, Nati Daniel, and Yoli Shavit. JEPA-DNA: Grounding genomic foundation models through joint-embedding predictive architectures. *arXiv preprint arXiv:2602.17162*, 2026. URL https://arxiv.org/abs/2602.17162.

Yann LeCun. A path towards autonomous machine intelligence. *OpenReview preprint*, 2022. URL https://openreview.net/forum?id=BZ5a1r-kVsf.

Etai Littwin, Omid Saremi, Madhu Advani, Vimal Thilak, Preetum Nakkiran, Chen Huang, and Joshua Susskind. How JEPA avoids noisy features: The implicit bias of deep linear self distillation networks. In *Advances in Neural Information Processing Systems*, volume 37, pp. 91300–91336. Curran Associates, Inc., 2024. doi: 10.52202/079017-2897. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/a600f0f740605205133553cb74a1c107-Abstract-Conference.html.

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. LeWorldModel: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2026. URL https://arxiv.org/abs/2603.19312.

Léo Nicollier, Max Dunitz, Marc Pic, Pablo Musé, Enric Meinhardt-Llopis, and Gabriele Facciolo. SPHERE-JEPA: Spherical prediction with homogeneous embeddings. *arXiv preprint arXiv:2605.26900*, 2026. URL https://arxiv.org/abs/2605.26900.

Stephanie E. Palmer, Olivier Marre, Michael J. Berry, II, and William Bialek. Predictive information in a sensory population. *Proceedings of the National Academy of Sciences*, 112(22):6908–6913, 2015. doi: 10.1073/pnas.1506855112.

Ayan Pendharkar. Predictive objectives discard exogenous control-relevant features: A controlled mechanistic study. *arXiv preprint arXiv:2606.30068*, 2026. URL https://arxiv.org/abs/2606.30068.

Pratyaksh Rao, Wancong Zhang, Randall Balestriero, Yann LeCun, and Giuseppe Loianno. Sky-JEPA: Learning long-horizon world models for zero-shot sim-to-real control of quadrotors. *arXiv preprint arXiv:2606.23444*, 2026. URL https://arxiv.org/abs/2606.23444.

Jürgen Schmidhuber. Making the world differentiable: On using self-supervised fully recurrent neural networks for dynamic reinforcement learning and planning in non-stationary environments. Technical Report FKI-126-90, Institut für Informatik, Technische Universität München, 1990. URL https://people.idsia.ch/~juergen/FKI-126-90ocr.pdf.

Yosef Singer, Luke Taylor, Ben D. B. Willmore, Andrew J. King, and Nicol S. Harper. Hierarchical temporal prediction captures motion processing along the visual pathway. *eLife*, 12:e52599, 2023. doi: 10.7554/eLife.52599.

Vlad Sobal, Jyothir S V, Siddhartha Jalagam, Nicolas Carion, Kyunghyun Cho, and Yann LeCun. Joint embedding predictive architectures focus on slow features. *arXiv preprint arXiv:2211.10831*, 2022. URL https://arxiv.org/abs/2211.10831. Presented at the NeurIPS 2022 Workshop on Self-Supervised Learning: Theory and Practice.

Richard S. Sutton. Dyna, an integrated architecture for learning, planning, and reacting. *ACM SIGART Bulletin*, 2(4):160–163, 1991. doi: 10.1145/122344.122377. URL https://doi.org/10.1145/122344.122377.

Enigma Team. The obsessed encoder, 2026. URL https://enigma.inc/posts/obsessed-encoder.

Aleksandar Vujinovic and Aleksandar Kovacevic. ACT-JEPA: Novel joint-embedding predictive architecture for efficient policy representation learning. *IEEE Access*, 14:78895–78906, 2026. doi: 10.1109/ACCESS.2026.3696039. URL https://doi.org/10.1109/ACCESS.2026.3696039.

Haiyu Wu, Randall Balestriero, and Morgan Levine. VISReg: Variance-invariance-sketching regularization for JEPA training. *arXiv preprint arXiv:2606.02572*, 2026. URL https://arxiv.org/abs/2606.02572.

Haoran Zhu and Anna Choromanska. Self-supervised JEPA-based world models for LiDAR occupancy completion and forecasting. *arXiv preprint arXiv:2602.12540*, 2026. URL https://arxiv.org/abs/2602.12540.

Penghao Zhu, Salvatore Penachio, Kaustav Mukherjee, and Aneesh Jonelagadda. Spectral-target physical latent structuring for JEPA-style world models. *arXiv preprint arXiv:2609.04264*, 2026. URL https://arxiv.org/abs/2609.04264.

A Formalization of Predictive Closure

Appendices A.1 and A.2 prove Equations (2) and (3) of Section 3, respectively. Throughout, the prescribed distribution P and latent dimension k are fixed across encoders.

A.1 Gaussian Prediction Errors

Prediction-error assumption. Write $Z_t = f(X_t)$. We assume that all encoders produce k-dimensional latents with the same marginal distribution P, whose Shannon entropy $H(P)$ is finite. For each encoder f, assume there is a predictor g_f in the predictor class such that, at the latent's numerical resolution,

$$Z_{t+1} \mid Z_t = z \sim \mathcal{N}(g_f(z), \sigma_f I_k)\,, \tag{4}$$

where $\sigma_f > 0$ is a scalar variance that may differ between encoders.

Proof. For each current latent, squared prediction error is minimized by the mean of this Gaussian, which g_f predicts. Each error coordinate has zero mean and variance σ_f, so its expected squared error is σ_f. Summing over the k coordinates gives

$$\mathcal{E}_f := \min_g \mathbb{E}\,\|Z_{t+1} - g(Z_t)\|_2^2 = k\sigma_f. \tag{5}$$

The conditional Gaussian distribution has entropy

$$h(Z_{t+1} \mid Z_t) = \frac{k}{2}\log(2\pi e\sigma_f). \tag{6}$$

At the latent's fixed numerical precision, we assume the discrete conditional entropy equals this Gaussian value plus an additive correction c shared across encoders. Since the next latent has the fixed marginal distribution P,

$$\begin{aligned} I(Z_t; Z_{t+1}) &= H(P) - H(Z_{t+1} \mid Z_t) \\ &= H(P) - \frac{k}{2}\log\left(\frac{2\pi e}{k}\mathcal{E}_f\right) - c. \end{aligned} \tag{7}$$

Because $H(P)$, k, and c are fixed, predictive information is strictly decreasing in the optimized prediction error. Therefore, minimizing expected squared prediction error and maximizing predictive information select exactly the same optimal encoders.

A.2 From Representations to Feature Sets

Reader condition. For $Z = f(X_t) \sim P$, we control reader sensitivity using common tolerances $\varepsilon, \delta > 0$ and $0 < \alpha < 1$, keeping these tolerances unchanged across encoders:

$$\Pr\left[\sup_{\|z'-Z\|_2 \le \varepsilon} |d(z') - d(Z)| > \delta\right] \le \alpha.$$

Thus, latent perturbations of size at most ε change the reader's output by at most δ, except on a set of probability at most α. We require exact recovery from the unperturbed latent.

Information preservation. The latent determines every feature value in $Y_t(S(f))$. Conversely, the readers $d_j(z) = (\delta/\varepsilon)z_j$, for $j = 1, \ldots, k$, satisfy the sensitivity condition everywhere, so the corresponding features $x \mapsto d_j(f(x))$ belong to $S(f)$. Their values recover every latent coordinate. Because the latent and the complete feature values can each be recovered from the other, neither description loses information. Their mutual information across consecutive observations is therefore equal:

$$I\big(f(X_t); f(X_{t+1})\big) = I\big(Y_t(S(f)); Y_{t+1}(S(f))\big). \tag{8}$$

Since every feasible set is $S(f)$ for some encoder f, this equality and Appendix A.1 together show that $S(f^\star)$ maximizes predictive information over feasible feature sets, proving Equation (3).

B Worst- and Best-Position Readouts for Group Size

We check whether the set switch in Figure 4 also appears at the worst and best positions, using the same frozen encoders and position-specific image-detection probes described in Appendix C.2. For each optimization seed, we first average accuracy across target identities at each position, then take the minimum or maximum across positions within each group. Both measures show the same endpoint reversal in all three seeds: the competing cycle is more strongly represented at $N = 2$, whereas the smaller A/B group is more strongly represented at $N = 16$. The average probability of copying an image from the preceding position in each cycle remains unchanged across these sizes.

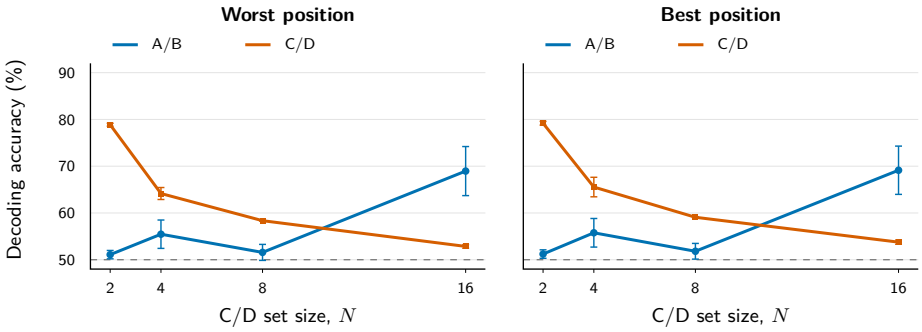


Figure 7: **Worst- and best-position readouts show the same set switch.** Both measures favor the competing cycle at the smallest size and A/B at the largest, showing that the reversal also holds for the least and most decodable positions within each group. Curves show means across three training runs; error bars show standard errors. Probes are trained on collages from images held out from encoder training and evaluated on separate collages. The dashed line marks the chance image-detection accuracy of 50%.

C/D accuracy decreases across all four sizes in every seed, but A/B accuracy is not monotonic, and the intermediate-size preferences vary across seeds. At $N = 16$, A/B decoding also varies more across optimization seeds. We hold the training and evaluation budget per position fixed across group sizes. Appendix C.2 describes the constant-area rendering procedure.

C Experimental Setup and Evaluation Details

C.1 Shared training and probing protocol

In each experiment in Section 4, we vary a single design parameter while keeping the remaining data-generation rules fixed. At each value of the varied parameter, we train the same end-to-end JEPA from scratch on sequences of collages composed of CIFAR-10 images, using a three-dimensional latent and the objective $0.9\mathcal{L}_{\text{pred}} + 0.1\mathcal{L}_{\text{SIGReg}}$ (Balestriero & LeCun, 2025). Training draws from a class-balanced set of 5,000 images from the official training split; after training, we freeze the encoder and train probes to decode designated features from newly sampled collages drawn from a disjoint class-balanced set of 5,000 test images.

Architecture and gradients. The encoder has four 3×3 convolution/ReLU/2×2 max-pooling blocks with 32, 64, 128, and 256 channels, followed by a $4096 \to 1024 \to 3$ projection with a GELU hidden activation. The group-size encoder additionally uses adaptive max-pooling to produce a 4×4 spatial output. Predictors use $3 \to 16 \to 16 \to 3$ GELU MLPs, with six inputs for side information; the reconstruction decoder is an affine $3 \to 48$ map. We share the main encoder across frames and backpropagate through both prediction-loss branches, without stop-gradient.

C.2 Set-switching setups and readouts

Changing predictability. The two-pair setup groups image identities into $\{A, B\}$ and $\{C, D\}$ at the four locations of a 2×2 collage. Within each pair, an image identity can be copied to the other position in the next frame: B_t determines A_{t+1} with probability 1.0, A_t determines B_{t+1} with probability 0.6, and D_t determines C_{t+1} with probability 0.9. The probability p_D that C_t determines D_{t+1} is the only parameter we vary; when a copy does not occur, the next image is sampled independently.

Changing group size. The A/B cycle is fixed at two identities with copy probability 0.85, and the competing C/D cycle has $N \in \{2, 4, 8, 16\}$ identities, ordered $C, D, E, F, \ldots$ through the first N positions, with copy probabilities alternating between 0.90 and 0.96. Thus the average per-feature copy probabilities remain 0.85 for A/B and 0.93 for C/D at every size. We train three runs per configuration, each for 100 epochs of 100,000 eight-frame sequences, with batch size 512 and a three-dimensional latent. Within each design, $\{A, B\}$ remains unchanged, allowing us to assess whether changes in the competing group alter its representation. We used the first run during configuration development and kept the configuration unchanged for the two replications; all three runs share the image pools and data-generation randomness.

Rendering control. Keeping individual image size fixed would make the larger C/D group occupy more of the input, confounding feature count with the group's total spatial extent. We therefore keep the 128×256-pixel canvas and the two native 32×32-pixel A/B images fixed, while distributing approximately the same total displayed area among the C/D images. To avoid removing image detail through downsampling, we use the native CIFAR-10 resolution as the minimum tile size and enlarge images when fewer positions share this area. For $N = 2, 4, 8, 16$, individual C/D images have side lengths of 90, 64, 45, and 32 pixels, respectively, giving occupied areas of 16,200, 16,384, 16,200, and 16,384 pixels, which differ by less than 1.2%. The C/D images occupy centered row-major grids of 1×2, 2×2, 2×4, and 4×4 within the same right-hand region; unused pixels remain black. Larger tiles use bilinear enlargement of the native CIFAR-10 images, without cropping or reducing any image below its original resolution. Enlargement is greatest at $N = 2$ and goes back to the original size at $N = 16$. Tile scale and layout still change with N, so this control does not isolate representation cost from the difficulty of extracting the images' information.

Image-detection probes. To assess image-identity information in the latent, we train target-image presence detectors separately for each fixed position. The group-size experiment uses the same primary image-detection probe architecture, per-detector training budget, and held-out image pool as the four-image predictability and steering experiments. We train a separate binary classifier for each of 50 target images at every fixed position, with five targets per CIFAR-10 class. Each classifier receives only the frozen latent representation: positive examples contain its target at the designated

position, while negative examples exclude that target everywhere and use a different image of the same class at the designated position. Other positions are sampled independently, excluding the target. Because negative examples exclude the target everywhere, a representation of the unordered images within a pair could pass both position-associated probes without distinguishing their positions. These readouts measure target-image detection under the stated sampling protocol and do not establish position binding.

Each detector has two hidden layers of width 16 and trains for 1,000 steps with batch size 128 on 512 training examples, with 512 validation and 1,024 test examples, balanced between positive and negative cases. We use AdamW with learning rate 10^{-3} decaying to 10^{-5}, 50 warmup steps, weight decay 10^{-4}, gradient clipping at 10, and the same probe optimization seed across comparisons. Latent coordinates are standardized using 100,000 separately sampled collages from the held-out image pool. All probe images are unseen during encoder training; target identities recur across probe training, validation, and testing, while the collages are sampled separately from the same held-out pool. We report ordinary binary classification accuracy at the zero-logit threshold, with chance accuracy $1/2$. Increasing the number of positions increases the number of independent detectors, rather than reducing the training or evaluation budget per position. We first average accuracy over target images within each position, then average positions within each group, and finally report the mean and standard error across optimization seeds.

C.3 Steering interventions

Both interventions use the two-pair setup, keeping the sequence's predictive relations, image pool, three-dimensional main latent representation, and training budget unchanged. Held-out image-detection probes receive only the frozen main encoder's representation.

Auxiliary reconstruction. To directly reward retaining information about C, we add an auxiliary reconstruction objective. The decoder reconstructs a standardized 4×4 average-pooled version of the current C image from the current latent representation, giving the objective $0.9\mathcal{L}_{\text{pred}} + 0.1\mathcal{L}_{\text{SIGReg}} + \beta\mathcal{L}^{C}_{\text{recon}}$. Only C is a reconstruction target; the decoder is discarded before probing. The four nonzero weights were identified in a preliminary search at $p_D = 0.75$ with both C and D as reconstruction targets. We then evaluate reconstruction of C alone at $\beta \in \{0, 0.0425, 0.105, 0.135, 0.36\}$ and all five original values of p_D, using one optimization seed.

Predictor side information. To examine how direct predictor access to C changes the main representation, we supply the current C image through a separate encoder. The side encoder has the same architecture as the main encoder and maps the current 32×32-pixel C image, bilinearly enlarged to 64×64 pixels, to a three-dimensional vector. We concatenate this vector with the current main latent to form the predictor's six-dimensional input; its target remains the next-frame main latent. The main encoder continues to receive the full collage, including C, and retains its three-dimensional output. Training uses $0.9\mathcal{L}_{\text{pred}} + 0.1\mathcal{L}_{\text{SIGReg}}$, with the side encoder trained through the prediction loss and SIGReg applied only to the main latent. This intervention adds encoder capacity while keeping the main latent dimension fixed. We evaluate all five values of p_D with three training runs per condition, paired with their corresponding no-side-input controls.

D REGULARIZATION SENSITIVITY

We test how the findings change with the regularization weight in all five experiments, using

$$\mathcal{L}_r = (1-r)\mathcal{L}_{\text{pred}} + r\mathcal{L}_{\text{SIGReg}}, \qquad r \in \{0.05, 0.10, 0.20\}. \tag{9}$$

The main setting uses $r = 0.10$; the corresponding weights relative to a unit prediction coefficient are $\lambda = r/(1-r) \in \{1/19, 1/9, 1/4\}$. For reconstruction, we add $\beta\mathcal{L}_{\text{rec},C}$ and hold β fixed across r. All other training and probing settings follow Appendix C. Figures 8–12 show every tested condition. We report mean decoding accuracy with standard errors across three runs, except for reconstruction, which uses one run.

Across the tested λ values, we recover the qualitative findings reported in the main text for cross-subsidization, predictability-driven set switching, and steering through reconstruction or side information. The group-size result holds at the larger λ; at the smaller λ, the enlarged group still becomes less represented, but the switch in favor of the competing group occurs in only one of three runs.

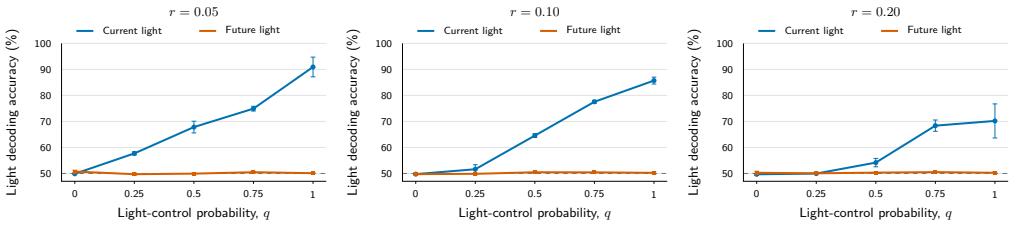


Figure 8: **Cross-subsidization.** The current light becomes more strongly represented as q increases, although stronger regularization weakens this effect; future-light decoding stays near chance. The future-light probe decodes L_{t+1} from the frozen current-frame latent $f(X_t)$. Dashed lines mark 50% chance light-decoding accuracy.

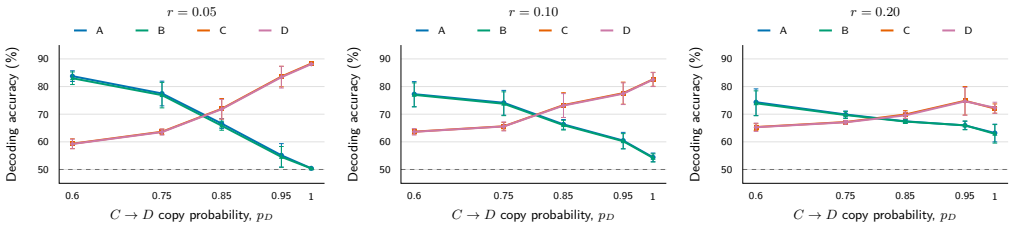


Figure 9: **Changing predictability.** Increasing p_D from 0.60 to 1 changes the better-represented pair from A/B to C/D in all three runs at every regularization weight. Stronger regularization reduces the separation between the pairs. The change need not be monotonic: at $r = 0.20$, mean decoding accuracy for C/D decreases from $p_D = 0.95$ to $p_D = 1$. Dashed lines mark 50% chance accuracy.

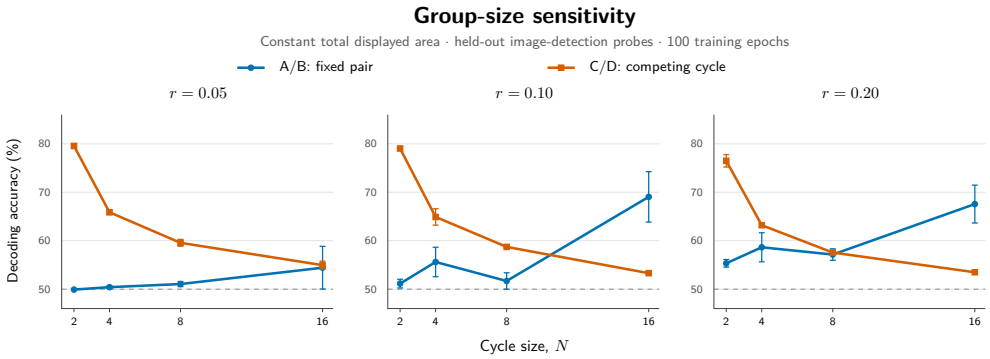


Figure 10: **Changing group size.** Enlarging the C/D group reduces its decoding accuracy in every run at every regularization weight. At $N = 16$, A/B becomes better represented in all three runs at $r = 0.10, 0.20$, but only one of three at $r = 0.05$. The relative strength of the two groups need not change monotonically with N. We average decoding accuracy across positions within each group; dashed lines mark 50% chance accuracy.

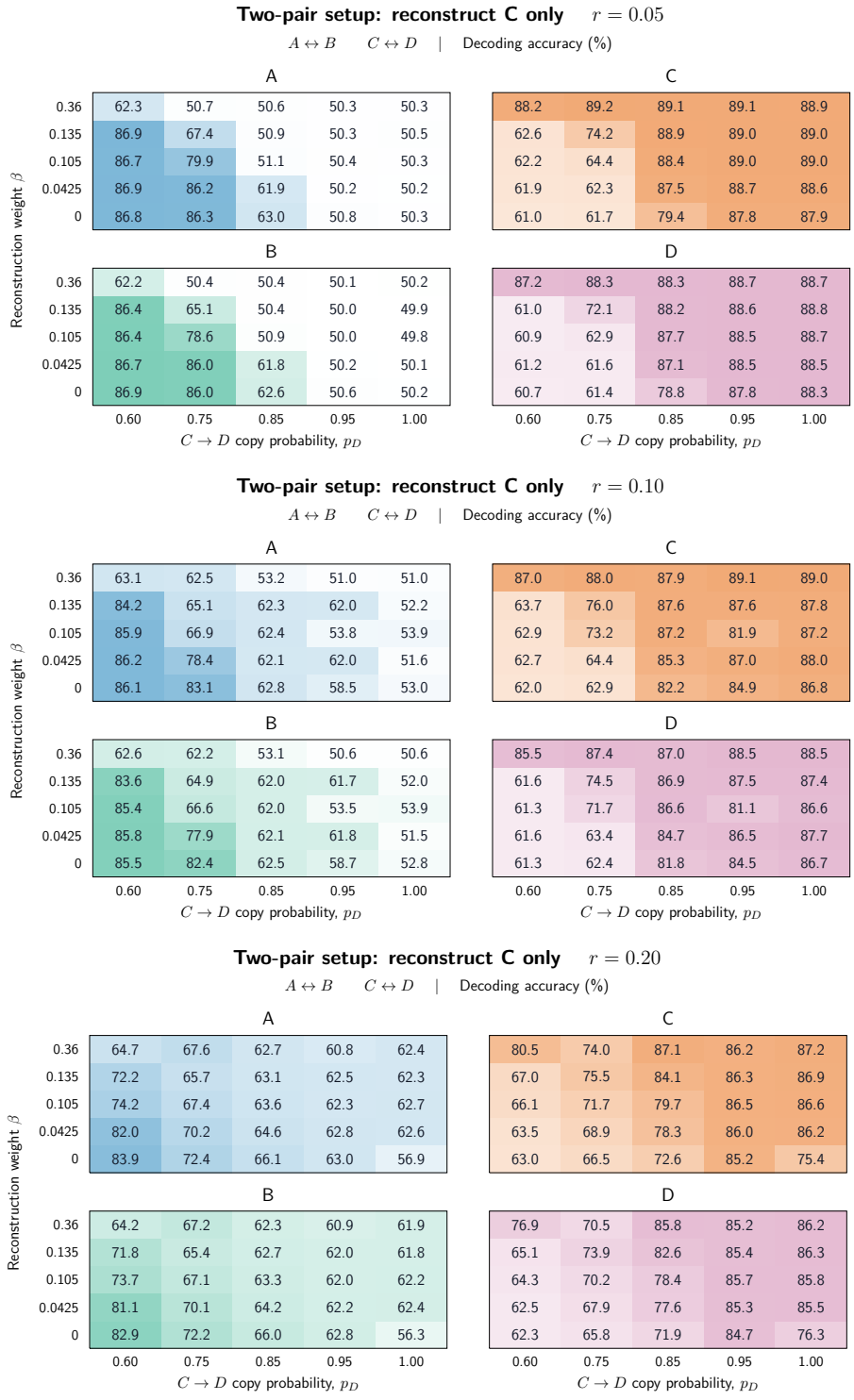


Figure 11: **Auxiliary reconstruction.** Reconstructing C can also strengthen its predictive partner D, which receives no reconstruction supervision. At $\beta = 0.36$, C/D is better represented than A/B at every p_D and regularization weight. Increasing β does not always improve decoding: at $r = 0.20, p_D = 0.75$, both C and D decline from $\beta = 0.135$ to $\beta = 0.36$. All four images remain visible to the encoder. Cells show decoding accuracy from one training run, using a shared 50–90% color scale; numerical values below 50% remain visible.

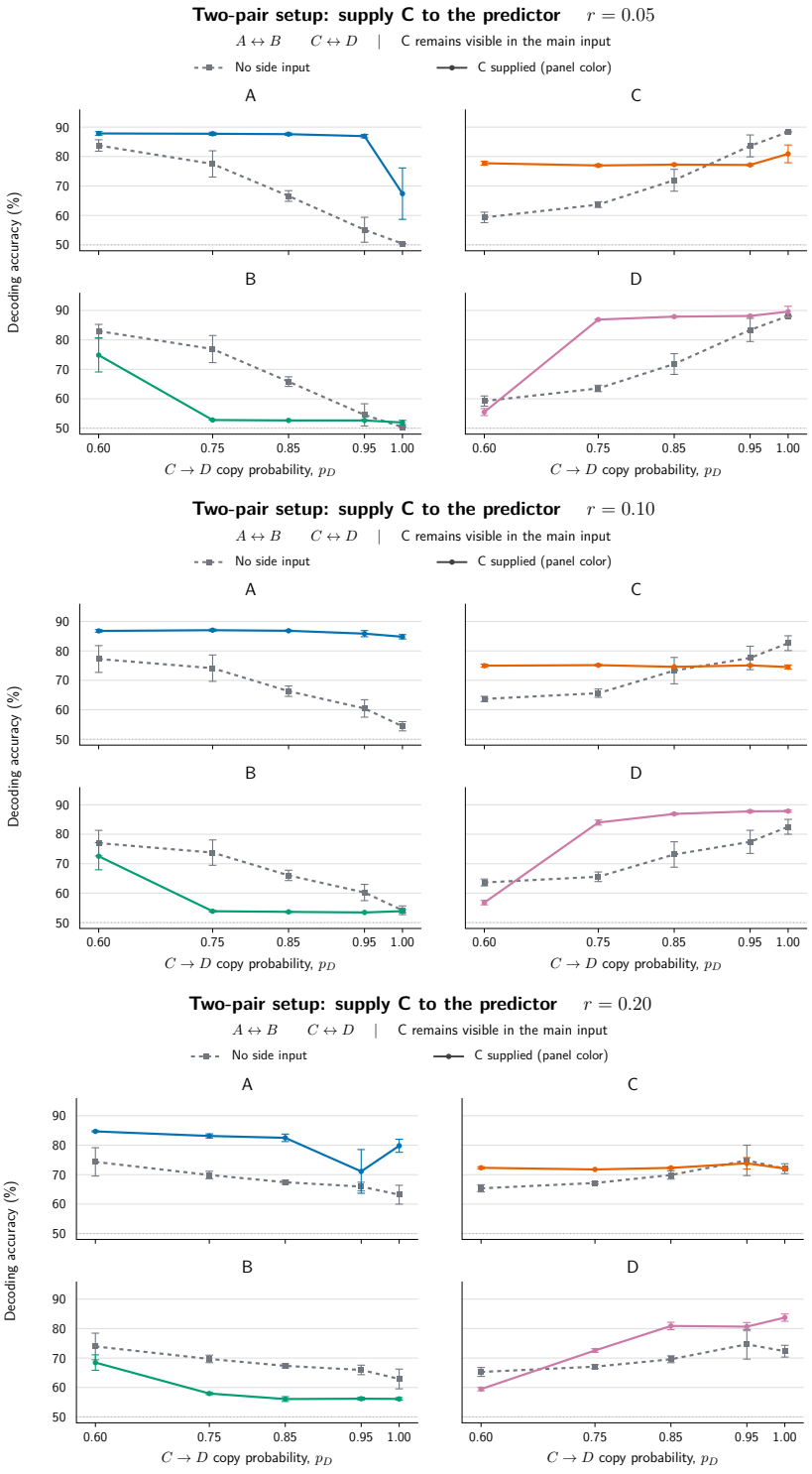


Figure 12: **Predictor side information.** Supplying C strengthens representation of A and D while weakening B at $p_D = 0.75, 0.85$ in all three runs at every regularization weight. The effect varies across the sweep and does not uniformly replace the A/B pair with C/D. Colored curves supply C to the predictor; gray curves omit side input. All probes read the main latent without access to the side input; supplying C also adds encoder capacity. Panels show decoding accuracy with standard errors across three runs; dotted lines mark 50% chance accuracy.

E NEXT-OBSERVATION PREDICTION

To validate that the JEPA systems learned to predict, we test whether their predictions identify the actual next observation among two candidates. We evaluate the traffic-light setup at $q = 1$ and the two-pair setup at all five values of p_D, with and without side information. Training settings follow Appendix C.

We evaluate each trained system on $5{,}000$ transitions generated from images held out from encoder training, using the same transitions across runs. For each transition, we compare the predicted next latent with the main encoder's representations of the actual next observation and an alternative sampled using the sequence's transition rules. The alternative is equally likely as an observation overall, but cannot follow the given current observation. We count the prediction as correct if it is closer to the actual next observation's representation in squared Euclidean distance. As a control, we shuffle predictions across transitions while keeping the candidate observations fixed.

In every tested setting, the systems identify the actual next observation well above chance, whereas shuffled predictions remain near chance (Table 1).

Table 1: **Predictors distinguish actual next observations from incompatible alternatives.** Values are percentage correct, reported as mean $\pm$ standard error across three training runs. Chance is 50% for choosing the actual next observation from the two candidates. Each system is evaluated in its own learned latent space. The side-information condition supplies the current C image while keeping it visible in the main input.

	No side information		Side information	
Setting	Prediction	Shuffled	Prediction	Shuffled
Traffic light ($q = 1$)	77.13 ± 0.51	49.97 ± 0.43	—	—
Two pairs, $p_D = 0.60$	88.88 ± 0.35	50.10 ± 0.38	91.27 ± 0.19	49.95 ± 0.28
Two pairs, $p_D = 0.75$	90.65 ± 0.14	50.27 ± 0.23	93.64 ± 0.04	50.13 ± 0.11
Two pairs, $p_D = 0.85$	92.22 ± 0.08	50.56 ± 0.16	95.39 ± 0.07	49.70 ± 0.06
Two pairs, $p_D = 0.95$	94.53 ± 0.36	49.83 ± 0.33	97.15 ± 0.04	49.59 ± 0.07
Two pairs, $p_D = 1.00$	96.99 ± 0.49	50.29 ± 0.40	97.96 ± 0.13	49.83 ± 0.23